--Creearea bazei de date

create database dbSupermarket

on

(

name=supermarket,

filename='D:\Baze date\Supermarket\supermarket.mdf'

)

use dbSupermarket

--Creearea tabelelor

create table tProduse

(codProd int identity(1,1) constraint pk\_Prod primary key,

denumire varchar(60) not null,

);

create table tFurnizori

(codFurn int identity(1,1) constraint pk\_Furn primary key,

denumire varchar(60) not null

);

alter table tProduse

add codFurn int constraint fk\_Furn foreign key references tFurnizori(codFurn)

alter table tProdFact

drop column data

create table tAngajati

(codAngajat int identity(1,1) constraint pk\_Ang primary key,

nume varchar(60) not null,

CNP char(13) unique not null,

localitate varchar(20),

);

create table tDepartamente

(codDep int identity(1,1) constraint pk\_Dep primary key,

denumire varchar(60) not null,

);

create table tFacturi

(codFactura int identity(1,1) constraint pk\_Fac primary key,

data smalldatetime not null,

);

create table tProdFact

(codFactura int constraint fk\_cf foreign key references tFacturi(codFactura),

codProd int constraint fk\_cp foreign key references tProduse(codProd),

cantitate int not null,

data smalldatetime not null

constraint pk\_PF primary key(codFactura,codProd)

);

--Comanda INSERT

insert into tAngajati

values ('A', '0000000000001','Pitesti'),

('B', '0000000000002','Pitesti'),

('C', '0000000000003','Bucuresti'),

('D', '0000000000004','Bucuresti');

insert into tDepartamente

values ('IT'),

('Resurse umane'),

('Constructii'),

('Electrica'),

('Management');

insert into tFacturi

values ('20020118 11:23:09'),

('20030118 17:25:15'),

('20100623 11:23:09'),

('20120708 14:53:01');

insert into tFurnizori

values ('Furnizor1'),

('Furnizor2'),

('Furnizor3'),

('Furnizor4'),

('Furnizor5'),

('Furnizor6'),

('Furnizor7');

insert into tProduse

values ('Produs1','8'),

('Produs2','9'),

('Produs3','10'),

('Produs4','11'),

('Produs5','12');

insert into tProduse

values ('Produs6','12'),

('Produs7','11'),

('Produs8','10'),

('Produs9','9'),

('Produs10','8');

insert into tProdFact

values ('2','3',20),

('2','4',10),

('3','5',15),

('4','6',30),

('4','5',2);

insert into tProdFact

values ('5','2',30),

('5','3',10),

('5','4',15),

('5','5',25),

('5','6',20),

('6','2',40),

('6','3',25),

('6','4',5),

('6','5',10),

('6','6',30);

insert into tFacturi

values ('20110507 13:20:00');

--Comanda SELECT

select \* from tAngajati

select \* from tDepartamente

select \* from tFurnizori

select \* from tProduse

select \* from tFacturi

select \* from tProdFact

--Vreau sa adaug cheie secundara in tabelul tAngajati, codDep din tDepartamente

alter table tAngajati

add codDep int constraint fk\_Dep foreign key references tDepartamente(codDep)

update tAngajati

set codDep=1

where localitate='Pitesti'

update tAngajati

set codDep=2

where localitate='Bucuresti'

update tAngajati

set codDep=3

where localitate='Cluj'

update tAngajati

set codDep=4

where localitate='Baia Mare'

--afisati primii 2 angajati din tAngajati

select all top 2 \* from tAngajati

--afisati angajatii care locuiesc in Pitesti

select \* from tAngajati

where localitate='Pitesti'

--afisati numarul de angajati pe orase

select localitate, count(codAngajat) as "Numar angajati" from tAngajati

group by localitate

--afisati numarul de angajati pe orase in ordine crescatoare dupa "Numar angajati"

select localitate, count(codAngajat) as "Numar angajati" from tAngajati

group by localitate

order by "Numar angajati"

--afisati codurile produselor cu o cantitate totala vanduta peste 65

select codProd,sum(cantitate) as "Cantitate totala" from tProdFact

group by codProd

having sum(cantitate)>65

order by "Cantitate totala"

--Interogarea datelor din mai multe tabele

--Sa se determine furnizorii tuturor produselor

select tProduse.denumire as "Produs",tFurnizori.denumire as "Furnizor" from tFurnizori

inner join tProduse on tFurnizori.codFurn=tProduse.codFurn;

--afisati numele produselor co o cantitate totala vanduta peste 25

select tProduse.denumire,sum(cantitate) as "Cantitate totala" from tProdFact

inner join tProduse on tProdFact.codProd=tProduse.codProd

group by tProduse.denumire

having sum(cantitate)>25

order by "Cantitate totala"

--afisati toate produsele si furnizorii corespunzatori (right join)

select tProduse.denumire as "Produs",tFurnizori.denumire as "Furnizor" from tFurnizori

right join tProduse on tFurnizori.codFurn=tProduse.codFurn;

--afisati toti furnizorii si produsele furnizate (left join)

select tFurnizori.denumire as "Furnizor",tProduse.denumire as "Produs" from tFurnizori

left join tProduse on tFurnizori.codFurn=tProduse.codFurn;

--Subinterogari

--sa se afiseze angajatii aflati la departamentul 'IT'

select codAngajat, nume, A.codDep from tAngajati as A

inner join tDepartamente as B on A.codDep=B.codDep

where B.codDep=(select codDep from tDepartamente where denumire='IT')

--introducem date cu diferite orase si departamentul IT (codDep=1)

insert into tAngajati

values ('I','0000000000009','Craiova',1),

('J','0000000000010','Ploiesti',1),

('K','0000000000011','Timisoara',1),

('L','0000000000012','Sibiu',1);

--afisati toate orasele angajatilor de la departamentul IT

select distinct localitate from tAngajati

inner join tDepartamente on tAngajati.codDep=tDepartamente.codDep

where tAngajati.codDep=(select codDep from tDepartamente where denumire='IT')

--afisati produsele vandute intr-o cantitate mai mare decat produsul cu codul 5

select codProd,sum(cantitate) from tProdFact

group by codProd

having sum(cantitate)>(select sum(cantitate) from tProdFact where codProd='5')

alter table tAngajati

add salariu int

update tAngajati

set salariu=1500

where codAngajat=1

update tAngajati

set salariu=2500

where codAngajat=2

update tAngajati

set salariu=3500

where codAngajat=3

update tAngajati

set salariu=3000

where codAngajat=4

update tAngajati

set salariu=2900

where codAngajat=5

update tAngajati

set salariu=6000

where codAngajat=6

update tAngajati

set salariu=7500

where codAngajat=7

update tAngajati

set salariu=2800

where codAngajat=8

update tAngajati

set salariu=8000

where codAngajat=9

update tAngajati

set salariu=35000

where codAngajat=10

update tAngajati

set salariu=6800

where codAngajat=11

update tAngajati

set salariu=3200

where codAngajat=12

--sa se determine la nivel de departamente salariul mediu folosind subinterogare in clauza FROM

select denumire,Salariu\_mediu from tDepartamente

inner join (select codDep,avg(salariu) as Salariu\_mediu from tAngajati group by codDep) as A on tDepartamente.codDep=A.codDep

--Functia CASE

--sa se determine numarul salariilor de peste 4000 ron si numarul salariilor de sub 4000 de ron

select count(case when tAngajati.salariu<4000 then 1 end) as 'Sub 4000',

count(case when tAngajati.salariu>4000 then 1 end) as 'Peste 4000'

from tAngajati

--operatorul Union

alter table tFurnizori

add localitate varchar(20)

update tFurnizori

set localitate='Pitesti'

where codFurn=8

update tFurnizori

set localitate='Oradea'

where codFurn=9

update tFurnizori

set localitate='Sighisoara'

where codFurn=10

update tFurnizori

set localitate='Alba Iulia'

where codFurn=11

update tFurnizori

set localitate='Oradea'

where codFurn=12

update tFurnizori

set localitate='Alba Iulia'

where codFurn=13

update tFurnizori

set localitate='Constanta'

where codFurn=14

--Afisati orasele angajatilor si ale furnizorilor intr-o singura tabela

select localitate from tAngajati

union

select localitate from tFurnizori

select localitate from tAngajati

intersect

select localitate from tFurnizori

select localitate from tAngajati

except

select localitate from tFurnizori

--operatorul ROLLUP

--afisati salariul mediu per localitate si departament

select localitate,codDep, avg(salariu) as 'Salariu mediu'

from tAngajati

group by rollup(localitate, codDep)

--afisati departamentul, angajatul si salariul acestuia, cu un total al salariilor pe fiecare departament, iar apoi salariul total al tuturor angajatilor

select tDepartamente.denumire,nume, sum(salariu) from tAngajati

inner join tDepartamente on tDepartamente.codDep=tAngajati.codDep

group by rollup(tDepartamente.denumire, nume)

--TABELE PIVOT

--sa se afiseze salariile angajatilor din Pitesti si Bucuresti

select codAngajat,[Pitesti],[Bucuresti]

from (select localitate, codAngajat,salariu from tAngajati) as A

pivot (max(salariu) for localitate in ([Pitesti],[Bucuresti])) as B

--sa se afiseze cantitatea totala de produse vanduta pentru furnizorii 9 si 11

select [9],[11]

from (select codFurn,cantitate from tProduse

inner join tProdFact on tProduse.codProd=tProdFact.codProd) as A

pivot(sum(cantitate) for codFurn in ([9],[11])) as B

--limbajul de manipulare a datelor (insert,update,delete,truncate)

--creati un tabel independent de baza de date actuala introduceti date, modificatile, stergetile in functie de o conditie, iar apoi stergetile pe toate

create table tExercitiu

(id int constraint pk\_ID primary key,

numar int not null

);

insert into tExercitiu

select codProd,codFurn from tProduse

update tExercitiu

set numar=1000

where numar=12

delete from tExercitiu

where numar=1000

truncate table tExercitiu

drop table tExercitiu

--UTILIZAREA VEDERILOR

--creati o vedere care sa contina cantitatea de produse vandute

create view vCantitateVanduta (produs, cantitate\_vanduta)

as

select denumire, sum(cantitate) from tProdFact

inner join tProduse on tProdFact.codProd=tProduse.codProd

group by denumire

--testarea a cateva comenzi de influentare a datelor din view

delete from vCantitateVanduta

where produs='Produs5'

update vCantitateVanduta

set produs='Produs2'

where produs='Produs1'

--crearea unei vederi cu salariul angajatilor

create view vSalariiAngajati

as

select codAngajat,salariu from tAngajati

--incercarea de update

update vSalariiAngajati

set salariu=3000

where codAngajat=2

--stergerea si recreerea vederii cu o conditie de existenta a campului salariu

drop view vSalariiAngajati

create view vSalariiAngajati

as

select codAngajat,salariu from tAngajati where salariu>5000

update vSalariiAngajati

set salariu=3000

where codAngajat=9

--recreerea vederii cu WITH CHECK OPTION si testarea functionalitatii

drop view vSalariiAngajati

create view vSalariiAngajati

as

select codAngajat,salariu from tAngajati where salariu>5000

WITH CHECK OPTION

update vSalariiAngajati

set salariu=3000

where codAngajat=7

drop view vSalariiAngajati

drop view vCantitateVanduta

--INDEXAREA TABELELOR

create index index\_test on tAngajati(salariu)

select salariu from tAngajati

drop index tAngajati.index\_test;

select \* from tAngajati